

✔ Congratulations! You passed!

Grade received 100%

To pass 80% or higher

Go to next item

Assess Your Knowledge

Latest Submission Grade 100%

1. Which of the tables in the employees database contains data of **both** the employees and the department of the employee?

1 / 1 point

- ☐ employees
- ☐ dept\_manager
- ☒ dept\_emp
- ☐ departments

✔ Correct

Correct! The **dept\_emp table** contains data about the employees and the employee's departments. It has 4 columns and 13042 rows.

2. Which of the following can a subquery return? (*Select all that apply*)

1 / 1 point

- ☒ An entire table
- ✔ Correct

Correct! A subquery may return a single value (a scalar), a single row, a single column, or an entire table
- ☒ A single column
- ✔ Correct

Correct! A subquery may return a single value (a scalar), a single row, a single column, or an entire table
- ☒ A single value
- ✔ Correct

Correct! A subquery may return a single value (a scalar), a single row, a single column, or an entire table
- ☒ A single row
- ✔ Correct

Correct! A subquery may return a single value (a scalar), a single row, a single column, or an entire table

3. The BETWEEN operator cannot be used with a subquery

1 / 1 point

- ☐ False
- ☐ Maybe
- ☒ True
- ✔ Correct

Correct! **The BETWEEN operator cannot be used with a subquery.** However, the BETWEEN operator can be used within the subquery.

4. Write a query to retrieve **only** the customer\_id and the name of the customer that has purchased the most from the store (that is, for customers where the sales column is the highest)

1 / 1 point

☐

```
1 SELECT c.customer_id, c.customer_name
2 FROM customers c,
3      (SELECT customer_id, COUNT(*) AS cust_count
4        FROM sales
5        GROUP BY customer_id
6        ORDER BY cust_count DESC) AS a
7 WHERE c.customer_id = a.customer_id
8 ORDER BY a.cust_count DESC;
```

☒

```
1 SELECT c.customer_id, c.customer_name
2 FROM customers c,
3      (SELECT customer_id, COUNT(*) AS cust_count
4        FROM sales
5        GROUP BY customer_id
6        ORDER BY cust_count DESC
7        LIMIT 1) AS a
8 WHERE c.customer_id = a.customer_id
9 ORDER BY a.cust_count DESC;
```

☐

```
1 SELECT c.customer_id, c.customer_name, a.cust_count
2 FROM customers c,
3      (SELECT customer_id, COUNT(*) AS cust_count
4        FROM sales
5        GROUP BY customer_id
6        ORDER BY cust_count DESC
7        LIMIT 1) AS a
8 WHERE c.customer_id = a.customer_id
9 ORDER BY a.cust_count DESC;
```

✔ Correct

Correct! The above retrieves **only the customer\_id and the name of the customer** that has purchased the most from the store (that is, for customers where the sales column is the highest). The **LIMIT 1** inside the subquery helps to achieve this purpose.

5. It is more professional to apply the ORDER BY clause in the outer query.

1 / 1 point

- ☐ False
- ☒ True
- ☐ Not sure
- ✔ Correct

Correct! It is more professional to apply ORDER BY in the outer query

6. As an SQL user, how would you retrieve a list of all managers who became managers after the 1st of January, 1985 and are not in the Finance or HR department? (*Select all that apply*)

1 / 1 point

Here is a snippet of the departments table:

	dept_no	dept_name
	(PK) character (4)	character varying (40)
1	d001	Marketing
2	d002	Finance
3	d003	Human Resources
4	d004	Production
5	d005	Development
6	d006	Quality Management
7	d007	Sales
8	d008	Research
9	d009	Customer Service

Here is a snippet of the department managers (dept\_manager) table:

	emp_no	dept_no	from_date	to_date
	(PK) integer	(FK) character (4)	date	date
1	110022	d001	1985-01-01	1991-10-01
2	110039	d001	1991-10-01	9999-01-01
3	110085	d002	1985-01-01	1989-12-17
4	110114	d002	1989-12-17	9999-01-01
5	110183	d003	1985-01-01	1992-03-21
6	110228	d003	1992-03-21	9999-01-01
7	110303	d004	1985-01-01	1988-09-09
8	110344	d004	1988-09-09	1992-08-02
9	110386	d004	1992-08-02	1996-08-30
10	110420	d004	1996-08-30	9999-01-01
11	110511	d005	1985-01-01	1992-04-25

☐

```
1 SELECT * FROM dept_manager
2 WHERE from_date > '1985-01-01'
3 AND dept_no NOT IN (SELECT dept_no FROM departments
4                     WHERE dept_name NOT IN ('Finance','Human Resources'));
```

☒

1    SELECT \* FROM dept\_manager

2    WHERE from\_date > '1985-01-01'

3    AND dept\_no NOT IN (SELECT dept\_no FROM departments

4                        WHERE dept\_name = 'Finance' OR dept\_name = 'Human Resources');

☒ **Correct**

Correct! This query will correctly retrieve a list of all managers who became managers after the 1st of January, 1985 and **are not in** the Finance or HR departments.

☒

1    SELECT \* FROM dept\_manager

2    WHERE from\_date > '1985-01-01'

3    AND dept\_no NOT IN (SELECT dept\_no FROM departments

4                        WHERE dept\_name IN ('Finance', 'Human Resources'));

☒ **Correct**

Correct! This query will correctly retrieve a list of all managers who became managers after the 1st of January, 1985 and **are not in** the Finance or HR departments.

☐

1    SELECT \* FROM dept\_manager

2    WHERE from\_date > '1985-01-01'

3    AND dept\_no NOT IN (SELECT dept\_no FROM departments

4                        WHERE dept\_name = 'Finance' OR dept\_name = 'HR');

7. When you write a join statement to join tables in SQL, it is important to \_\_\_\_\_ *(Select all that apply)*

1 / 1 point

☒ join on the related or common column

☒ **Correct**

Correct! When you write a join statement to join two or more tables in SQL, it is very important to join on their common column. *To learn more about SQL joins, check out [Mastering SQL Joins](#)*

☒ join on the related or common field

☒ **Correct**

Correct! When you write a join statement to join two or more tables in SQL, it is very important to join on their common column. *To learn more about SQL joins, check out [Mastering SQL Joins](#)*

☐ join on the related or common record/row

☐ join on the related entity

8. Given the SQL code below:

1 / 1 point

9    ORDER BY emp\_no;

How would you return only ten (10) results in the result set?

☐ UPTO 10

☐ MAX 10

☒ LIMIT 10

☐ ONLY 10

☒ **Correct**

Correct! The SQL LIMIT statement is used to retrieve records from one or more tables in a database and limit the number of records returned based on a limit value.

9. In question 8 above, where will you add the **LIMIT clause** to the query?

1 / 1 point

☐ After the FROM clause before the JOIN keyword

☒ After the ORDER BY clause

☐ Before the ON keyword

☐ Before the ORDER BY clause

☒ **Correct**

Correct! The **LIMIT clause** is usually the last keyword in an SQL query. In this case, it will come after the ORDER BY clause.

10. The \_\_\_\_\_ operator tests for set membership, where the set is a collection of values produced by a SELECT clause. The \_\_\_\_\_ operator tests for the absence of set membership

1 / 1 point

☐ IN, OR

☐ OR, IN

☒ IN, NOT IN

☐ NOT IN, IN

☒ **Correct**

Correct! The **IN** operator tests for set membership, where the set is a collection of values produced by a SELECT clause. The **NOT IN** operator tests for the absence of set membership

https://www.coursera.org/learn/working-with-subqueries-in-sql/exam/FEJPQ/assess-your-knowledge/attempt?redirectToCover=true

2/309